Lab 10 - Iteration and Recursion - Analysis

Katrine Chow

CS 162 - Winter '17

For this lab, we are asked to implement and existing iterative and recursive algorithms that calculates Fibonacci numbers.
I referenced the "Fibonacci Recursive and Non Recursive C++" article per Professor Zhang's recommendation, found at:
https://www.codeproject.com/tips/109443/fibonacci-recursive-and-non-recursive-c
I modified the code to focus on the iterative and recursive portion, and in main.cpp, added clock (from <ctime>) functions to calculate and compare running times of the two functions.

Per Professor Zhang's comments in the Week 10 Overview section on Canvas, I have the impression that since Recursion counts as separate function calls on the stack, this type of function would be processed more slowly than iterations.
I expect to see slower clock speeds from the recursive function in this Lab.

Test and Comparison

Nth Fibonacci Number	Iteration Time (in ms)	Recursion Time (in ms)	
0	0	0	
1	0	0	
2	0	0	
10	0	0	
20	0	0	
45	0	17340	
50	0	191650	
51	0	408340	
55	0	N/A (taking a very long time)	
2,515,000	10	N/A (taking a very long time)	
n (when diff. first shows)	0	10	**n = 29**

Analysis

I capped the input range to 0 - 50 as numbers larger than 51 tends to take quite awhile to display.
For testing purpose, I tried to input a very large number to test iterative function's running time

I found that I do not see any significant clock time until n is above 2,515,000.
For recursive function however I first see a non-zero clock time when n reaches 29.
This very clearly shows that iterative functions run much faster than recursive functions.